**Name**: Modify traffic light system

**Goal**: Modify an existing traffic light system on the grid

**Actor**: User

**Pre-condition**: System is displaying a new or a loaded project so that the user can modify the grid and there must already be a crossing with a traffic light. System is in state “stopped” or “paused”.

**MSS**:

1. User hovers over a cell
2. System output a small icon in the top left corner
3. User clicks on the icon
4. System redirects user to traffic light and flow settings
5. User chooses to modify the interval times of the states of the traffic light system
6. System shows the user the current parameters
7. User changes the values of the inputs of the states
8. User clicks confirm
9. System modifies traffic light system with new parameters
10. System recalculates total interval time

**Exception (Extension, Alternatives)**:

7.1: User leaves input value blank

System modifies the traffic light with default value

7.2: User inputs wrong values

System outputs an appropriate message “You have to input values of type integer 1-100”

7.3: User sets the time to too high or too low

System notifies the user of the boundaries.

**Post-condition**: The selected traffic light has now changed parameters.

**Name**: Change traffic light setup(Optional)

**Goal**: Changes an existing traffic light system on the grid to another one

**Actor**: User

**Pre-condition**: System is displaying a new or a loaded project so that the user can modify the grid and there must already be a crossing with a traffic light. System is in state “stopped” or “paused”.

**MSS**:

1. User right clicks on a crossing
2. User chooses the “change traffic light system” option
3. System shows the user the traffic light setups the user can choose from
4. User selects a new set up for the traffic light system
5. System asks user to input values of states
6. User inputs values for each state
7. System replaces existing traffic light system with a new one

**Exception (Extension, Alternatives)**:

6.1: User leaves input value blank

System modifies the traffic light with default value

6.2: User inputs wrong values

System outputs an appropriate message “You have to input values of type integer 1-100”

6.3: User sets the time to too high or too low

System notifies the user of the boundaries.

**Post-condition**: The selected traffic light system has now been changed to another.

**Name**: Add crossing

**Goal**: Adds a crossing to our grid

**Actor**: User

**Pre-condition**: System is displaying a new or a loaded project so that the user can modify the grid. System is in state “paused” or “stopped”.

**MSS**:

1. User chooses a crossing he wants to add
2. User drags a crossing to the cell he wants to place it in
3. User drops the crossing in the cell
4. System redirects user to traffic light system setup
5. System asks user what kind of traffic light system he would like to choose(Optional)
6. User selects a traffic light system(Optional)
7. System asks user to input values of the traffic light system
8. User input values for each traffic light system state
9. System places the crossing on the grid
10. System adds new cell to the calculation of the flow

**Exception (Extension, Alternatives)**:

3.1: Cell already has a crossing on it -> Go to use case “Change crossing”

9.1: User doesn’t give inputs for traffic light system

System configures the traffic light system with default values

9.2: User inputs wrong values

System outputs an appropriate message “You have to input values of type integer 1-100”

9.3: User sets the time to too high or too low

System notifies the user of the boundaries

**Post-condition**: Our grid now has a new crossing on it.

**Name**: Delete crossing

**Goal**: Deletes an existing crossing from the grid

**Actor**: User

**Pre-condition**: System is displaying a new or a loaded project so that the user can modify the grid and there’s already a crossing on the grid to delete. System is in state “paused” or “stopped”.

**MSS**:

1. User right clicks on existing crossing on the map
2. User chooses the delete option
3. System asks the user for confirmation
4. The user confirms
5. System deletes crossing from the grid
6. System removes the cell from the calculation of the flow

**Exception (Extension, Alternatives)**:

5.1: Crossing was part of the path from start point to end point

System deletes navigation route

**Post-condition**: Our grid now has one less crossing

**Name**: Change crossing

**Goal**: Changes an existing crossing on the grid

**Actor**: User

**Pre-condition**: System is displaying a new or a loaded project so that the user can modify the grid and there’s already a crossing on the grid to change. System is in state “paused” or “stopped”.

**MSS**:

1. User right clicks on a cell with a crossing
2. User chooses the change option
3. System redirects user to crossing settings
4. User choose crossing he wants to place in said cell
5. System asks user to input values of the traffic light system
6. User input values for each traffic light system state
7. System removes previous crossing from the grid
8. System places the new crossing in the cell
9. System automatically changes the flow of the traffic

**Exception (Extension, Alternatives)**:

**Post-condition**: The grid now has a different kind of layout.

**Name**: Rotate crossing

**Goal**: Rotates an existing crossing on the map

**Actor**: User

**Pre-condition**: System is displaying a new or a loaded project so that the user can modify the map and there’s already a crossing on the map to rotate.

**MSS**:

1. User right clicks on existing crossing
2. User chooses the rotate option
3. System swaps the properties of the 4 roads with the properties of the one on the right.
4. System rotates the crossing picture(90 degrees counter clockwise)
5. System alters the flow appropriately

**Exception (Extension, Alternatives)**:

**Post-condition**: The map has now a different kind of layout.